APPENDIX A1

Motor Vehicles

```
<Ontology
    xmlns=http://www.abc.com/schemas/ontology.xsd
    ontologyOf=”motorCars”
    author=”John Smith”>


<classesDeclaration>

    <class classLabel=”Cars”>
        <classDescription>
            The motor vehicles we drive
        </classDescription>
    </class>

    <class classLabel=”CarManufacturers”>
        <classDescription>
            Corporations producing cars
        </classDescription>
        <subClassOf>
            <class classLabel=”LegalEntities”/>
        </subClassOf>
    </class>

    <class classLabel=”Models”>
        <classDescription>
            e.g. SuperLancer, A4
        </classDescription>
    </class>

    <class classLabel=”Persons”>
        <classDescription>
            instances of the species Homo Sapiens
        </classDescription>
        <subClassOf>
            <class classLabel=”LegalEntities”>
        </subClassOf>
    </class>

    <class classLabel=”LegalEntities”>
        <classDescription>
            Corporations and Persons
        </classDescription>
    </class>

    <class classLabel=”FuelTypes”>
        <classDescription>
            unleaded, leaded, 96 octane, etc.
        </classDescription>
    </class>

    <class classLabel=”TireTypes”>
        <classDescription>
            SKUs of various sorts of tires
        </classDescription>
    </class>

    <class classLabel=”Contracts”>
        <classDescription>
            those pieces of paper lawyers compose and review
        </classDescription>
    </class>

    <class classLabel=”TransmissionTypes”>
        <classDescription>
            Will that be standard or automatic?
        </classDescription>
    </class>

    <class classLabel=”BrakeSystems”>
        <classDescription>
            power brakes, ABS and others
        </classDescription>
    </class>

    <class classLabel=”EngineTypes”>
        <classDescription>
            V8, 5-cylinder and others
        </classDescription>
    </class>

    <class classLabel=”Distances”>
        <classDescription>
            miles or kilometers, or feet and inches, or meters
        </classDescription>
    </class>

    <class classLabel=”Speeds”>
        <classDescription>
            mph, kph, knots per second
        </classDescription>
    </class>

</classesDeclaration>


<relationsDeclaration>

    <relation relationLabel=owner>
        <domain>
            <class classLabel=”Cars” />
```

```
                <set>
                    <class classLabel=”LegalEntities”>
                </set>
            </domain>
        </relation>

        <relation relationLabel=insuranceCarrier>
            <domain>
                <crossProduct>
                    <class classLabel=”Cars” />
                    <set>
                    <class classLabel=”LegalEntities />
                    </set>
                </crossProduct>
                <class classLabel=”LegalEntities” />
            </domain>
        </relation>

        <relation relationLabel=insurancePolicy>
            <domain>
                <crossProduct>
                    <class classLabel=”Cars” />
                    <set>
                    <class classLabel=”LegalEntities />
                    </set>
                </crossProduct>
                <class classLabel=”Contracts” />
            </domain>
        </relation>

    </relationsDeclaration>


    <functionsDeclaration>

        <function functionLabel=make>

            <domain>
                <class classLabel=”Cars” />
            </domain>

            <range>
                <set>
                    <class classLabel=”CarManufacturers” />
                </set>
            </range>

        </function>

        <function functionLabel=fuelIntake>
            <domain>
                <class classLabel=”Cars” />
        </domain>
        <range>
            <class classLabel="FuelTypes" />
        </range>
    </function>

    <function functionLabel=color>
        <domain>
            <class classLabel="Cars" />
        </domain>
        <range>
            <set>
                <class classLabel="Colors" />
            </set>
        </range>
    </function>

    <function functionLabel=tires>
        <domain>
            <class classLabel="Cars" />
        </domain>
        <range>
            <bag>
                <class classLabel="TireTypes" />
            </bag>
        </range>
    </function>

    <function functionLabel=tireManufacturers>
        <domain>
            <class classLabel="TireTypes" />
        </domain>
        <range>
            <class classLabel="LegalEntities" />
        </range>
    </function>

    <function functionLabel=transmission>
        <domain>
            <class classLabel="Cars" />
        </domain>
        <range>
            <class classLabel="TransmissionTypes" />
        </range>
    </function>

    <function functionLabel=mileage>
        <domain>
            <class classLabel="Cars" />
        </domain>
        <range>
            <class classLabel="Distances" />
        </range>
    </function>

    <function functionLabel=maximumSpeed>
        <domain>
            <class classLabel="Cars" />
        </domain>
        <range>
            <class classLabel="Speeds" />
        </range>
    </function>

</functionsDeclaration>

</Ontology>
```

APPENDIX A2

Airline Travel

```
<Ontology
    xmlns=http://www.abc.com/schemas/ontology.xsd
    ontologyOf="Airline_Travel" >
    author="John Smith">


<classesDeclaration>

    <class classLabel="LegalEntities">
        <classDescription>
            Corporations and Persons
        </classDescription>
    </class>

    <class classLabel="Airlines">
        <classDescription>
            SABENA, BritishAir, El Al, TWA
    </classDescription>
        <subClassOf>
            <class classLabel="LegalEntities"/>
        </subClassOf>
    </class>

    <class classLabel="Airports">
        <classDescription>
            e.g. Heathrow, Gatwick, Dulles
        </classDescription>
    </class>

    <class classLabel="Persons">
        <classDescription>
            instances of the species Homo Sapiens
        </classDescription>
        <subClassOf>
            <class classLabel="LegalEntities">
        </subClassOf>
    </class>

    <class classLabel="Trips">
        <classDescription>
            going on a journey
        </classDescription>
    </class>

    <class classLabel="Locations">
        <classDescription>
            from your geography lessons
        </classDescription>
        </class>

        <class classLabel=”Flights”>
            <classDescription>
                flap your wings
            </classDescription>
        </class>

        <class classLabel=”AirplaneTypes”>
            <classDescription>
                Boeing 747, 777, DC-10, ….
            </classDescription>
        </class>

    </classesDeclaration>


    <relationsDeclaration>

        <relation relationLabel=travelers>
            <domain>
                <class classLabel=”Trips” />
                <set>
                    <class classLabel=”Persons”>
                </set>
            </domain>
        </relation>

        <relation relationLabel=destinations>
            <domain>
                <class classLabel=”Trips” />
                <set>
                    <class classLabel=”Locations” />
                </set>
            </domain>
    </relationsDeclaration>


    <functionsDeclaration>

        <function functionLabel=carrier>
            <domain>
                <class classLabel=”Flights” />
            </domain>
            <range>
                <class classLabel=”Airlines” />
            </range>
        </function>

        <function functionLabel=takeoff>
            <domain>
                <class classLabel=”Flights” />
        </domain>
        <range>
            <class classLabel=”Airports” />
        </range>
    </function>

    <function functionLabel=landing>
        <domain>
            <class classLabel=”Flights” />
        </domain>
        <range>
            <class classLabel=”Airports” />
        </range>
    </function>

    <function functionLabel=travelAgent>
        <domain>
            <class classLabel=”Trips” />
        </domain>
        <range>
            <set>
                <class classLabel=”LegalEntities” />
            </set>
        </range>
    </function>

    <function functionLabel=airplane>
        <domain>
            <class classLabel=”Flights” />
        </domain>
        <range>
            <class classLabel=”AirplaneTypes” />
        </range>
    </function>

    <function functionLabel=connectingFlight>
        <domain>
            <crossProduct>
                <class classLabel=”Trips” />
                <class classLabel=”Persons” />
                <class classLabel=”Flights” />
            </crossProduct>
        </domain>
        <range>
            <class classLabel=”Flights” />
        </range>
    </function>

</functionsDeclaration>

</Ontology>
```

APPENDIX A3

Purchase Order

```
<Ontology
    xmlns=http://www.abc.com/schemas/ontology.xsd
    ontologyOf=”Purchase_Orders”
    author=”John Smith”>


<classesDeclaration>

    <class classLabel=”LegalEntities”>
        <classDescription>
            Corporations and Persons
        </classDescription>
    </class>

    <class classLabel=”PurchaseOrders”>
        <classDescription>
            requests for purchases
        </classDescription>
    </class>

    <class classLabel=”Corporations”>
        <classDescription>
            incs, ltds, and so forth
        </classDescription>
        <subClassOf>
            <class classLabel=”LegalEntities”/>
        </subClassOf>
    </class>

    <class classLabel=”StockItems”>
        <classDescription>
            what is available for purchase
        </classDescription>
    </class>

    <class classLabel=”Persons”>
        <classDescription>
            instances of the species Homo Sapiens
        </classDescription>
        <subClassOf>
            <class classLabel=”LegalEntities”>
        </subClassOf>
    </class>

    <class classLabel=”Addresses”>
        <classDescription>
            e.g.,  47 Eden Street, Cambridge, postal code CB1 1JR
        </classDescription>
    </class>

</classesDeclaration>


<relationsDeclaration>

    <relation relationLabel=itemsPurchased>
        <domain>
            <class classLabel=”PurchaseOrder” />
            <set>
                <class classLabel=”StockItems”>
            </set>
        </domain>
    </relation>

    <relation relationLabel=providers>
        <domain>
            <class classLabel=”PurchaseOrder” />
            <set>
                <class classLabel=”LegalEntities” />
            </set>
        </domain>

</relationsDeclaration>


<functionsDeclaration>

    <function functionLabel=shipTo>
        <domain>
            <class classLabel=”PurchaseOrders” />
        </domain>
        <range>
            <class classLabel=”Addresses” />
        </range>
    </function>

    <function functionLabel=customers>
        <domain>
            <class classLabel=”PurchaseOrders” />
        </domain>
        <range>
            <set>
                <class classLabel=”LegalEntities” />
            </set>
        </range>
    </function>

    <function functionLabel=billTo>
        <domain>
            <class classLabel=”PurchaseOrders” />
            </domain>
            <range>
                <class classLabel="Addresses" />
            </range>
        </function>

</functionsDeclaration>

</Ontology>
```

APPENDIX B

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2000/10/XMLSchema" elementFormDefault="qualified">
    <xsd:annotation>
        <xsd:documentation>
        Ontology description schema
        Copyright 2001 Unicorn Solutions Inc. All rights reserved
        </xsd:documentation>
    </xsd:annotation>
    <!-- extraData group -->
    <xsd:group name="extraData">
        <xsd:choice>
            <xsd:element name="comment" type="comment"/>
            <xsd:element name="description" type="comment"/>
            <xsd:element name="example" type="comment"/>
            <xsd:element name="userLabel" type="comment"/>
            <!-- comment, description, example may be added freely -->
            <!-- userLabel: optional synonym. May only be used for reference external to the XML document -->
        </xsd:choice>
    </xsd:group>
    <!-- definition of classes type -->
    <xsd:complexType name="classes">
        <xsd:sequence>
            <xsd:element name="comment" type="comment" minOccurs="0" maxOccurs="unbounded"/>
            <xsd:element name="baseClass">
                <xsd:complexType>
                    <xsd:attribute name="label" type="xsd:ID" use="fixed" value="Being"/>
                </xsd:complexType>
                <!-- Being is the base class from which all other classes inherit. Always appears by default -->
            </xsd:element>
            <xsd:element name="class" minOccurs="0" maxOccurs="unbounded">
                <xsd:complexType>
            <xsd:sequence>
                <xsd:group ref="extraData" minOccurs="0" maxOccurs="unbounded"/>
                <!-- comments, descriptions, examples, userLabels relating to the class go here -->
            </xsd:sequence>
            <xsd:attribute name="label" type="xsd:ID" use="required"/>
            <xsd:attribute name="instanceLabel" type="xsd:string" use="optional"/>
            <xsd:attribute name="icon" type="xsd:string" use="optional"/>
            <xsd:attribute name="package" type="xsd:IDREF" use="optional"/>
            <!-- classLabel: the "official" ID label for the class to be used for all references within the XML document -->
            <!-- instanceLabel: optional label for instances of the class, if this is different from the label of the class -->
            <!-- icon: optional URL of link to icon, if such exists for class -->
        </xsd:complexType>
        <!-- this bloc is where the listing of user-defined classes appears -->
      </xsd:element>
    </xsd:sequence>
</xsd:complexType>
<!-- definition of classRef type -->
<xsd:complexType name="classRef">
    <xsd:attribute name="label" type="xsd:token" use="required"/>
    <xsd:attribute name="package" type="xsd:IDREF" use="optional"/>
    <!-- type for all references to classes. The references must relate to classLabels, not userLabels or instanceLabels -->
</xsd:complexType>
<!-- Explanation of complexClassType and complexClassListType:

    A complex class type is either a direct reference to a class ID - using the tag <classRef></classRef> -
    _or_ the tags <set></set> or <bag></bag> or <list></list> - with another complex class inserted recursively
    between those tags - _or_ the tags <intersection></intersection> or <Cartesian></Cartesian> - with a
    complex class list inserted between the tags.

    The complex class type is defined with reference to the complexClassGroup content grouping, which uses
     the "choice sequencing", meaning that only _one_ of the above tags may be used at each stage of the recursion.

    A complex class _list_, on the other hand, is defined to be a _sequence_ of complexClassGroup, and there
```

is no boundedness on the number of times a complex class tag may appear in such a list, nor is there any restriction on the order of the tags. It is therefore defined as a separate XML complex type.

```
-->
<xsd:group name="complexClassGroup">
    <xsd:choice>
        <xsd:element name="classRef" type="classRef"/>
        <xsd:element name="set" type="complexClassType"/>
        <xsd:element name="bag" type="complexClassType"/>
        <xsd:element name="list" type="complexClassType"/>
        <xsd:element name="sequence" type="complexClassType"/>
        <xsd:element name="intersection" type="complexClassListType"/>
        <xsd:element name="cartesian" type="complexClassListType"/>
    </xsd:choice>
</xsd:group>
<xsd:complexType name="complexClassListType">
    <xsd:sequence>
        <xsd:group ref="complexClassGroup" maxOccurs="unbounded"/>
    </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="complexClassType">
    <xsd:group ref="complexClassGroup"/>
</xsd:complexType>
<!-- definition of Complex Classes -->
<xsd:complexType name="complexClasses">
    <xsd:sequence>
        <xsd:element name="comment" type="comment" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="complexClass" minOccurs="0" maxOccurs="unbounded">
            <xsd:complexType>
                <xsd:sequence>
                    <xsd:sequence>
                        <xsd:group ref="extraData" minOccurs="0" maxOccurs="unbounded"/>
                        <!-- comments, descriptions, examples, userLabels relating to the complex class go here -->
                    </xsd:sequence>
                    <xsd:element name="component" type="complexClassType"/>
                    <!-- components is a wrapper for the main body of a complexClass -->
                    <!-- which is complecClassType -->
                </xsd:sequence>
                <xsd:attribute name="label" type="xsd:ID" use="required"/>
                <xsd:attribute name="package" type="xsd:IDREF" use="optional"/>
                <xsd:attribute name="instanceLabel" type="xsd:string" use="optional"/>
                <xsd:attribute name="icon" type="xsd:string" use="optional"/>
                <!--CCLabel:  the "official" ID for the complex class  to be used for all references within the XML document -->
                <!-- instanceLabel: optional label for instances of the complex class, if this is different from the label of the
                complexclass -->
                <!-- icon: optional URL of link to icon, if such exists for the complex class -->
            </xsd:complexType>
            <!-- here is where the listing of user-defined complex classes appears -->
        </xsd:element>
    </xsd:sequence>
</xsd:complexType>
<!-- definition of Relations type -->
<xsd:complexType name="relationType">
    <xsd:sequence>
        <xsd:sequence>
            <xsd:group ref="extraData" minOccurs="0" maxOccurs="unbounded"/>
            <!-- comments, descriptions, examples, userLabels relating to the relation go here -->
        </xsd:sequence>
        <xsd:element name="domain" type="complexClassType"/>
        <!-- this is where the domain of the relation appears. It is a complex class type -->
    </xsd:sequence>
    <xsd:attribute name="label" type="xsd:ID" use="required"/>
    <xsd:attribute name="package" type="xsd:IDREF" use="optional"/>
    <xsd:attribute name="icon" type="xsd:string" use="optional"/>
    <!-- relationLabel: the "official" ID label for the relation -->
    <!-- icon: optional URL of link to icon, if such exists for the relation -->
```

```
</xsd:complexType>
<xsd:complexType name="relations">
    <xsd:sequence>
        <xsd:element name="comment" type="comment" minOccurs="0"  maxOccurs="unbounded"/>
        <xsd:element name="relation" type="relationType" minOccurs="0" maxOccurs="unbounded"/>
        <!-- comment: over all comments on the list of relations - optional -->
    </xsd:sequence>
</xsd:complexType>
<!-- definition of Functions type -->
<xsd:complexType name="functionType">
    <xsd:sequence>
        <xsd:sequence>
            <xsd:group ref="extraData" minOccurs="0" maxOccurs="unbounded"/>
            <!-- comments, descriptions, examples, userLabels relating to the function go here -->
        </xsd:sequence>
        <xsd:element name="domain" type="complexClassType"/>
        <xsd:element name="co-domain" type="complexClassType"/>
        <!--domain: domain of function. It is a complex class type -->
        <!-- co-domain: also a complex class type -->
    </xsd:sequence>
    <xsd:attribute name="label" type="xsd:ID" use="required"/>
    <xsd:attribute name="package" type="xsd:IDREF" use="optional"/>
    <xsd:attribute name="icon" type="xsd:string" use="optional"/>
    <!--functionLabel:  the "official" ID label for the function  -->
    <!-- icon: optional URL of link to icon, if such exists for the function -->
</xsd:complexType>
<xsd:complexType name="functions">
    <xsd:sequence>
        <xsd:element name="comment" type="comment" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="function" type="functionType" minOccurs="0" maxOccurs="unbounded"/>
        <!-- comment: over all comments on the list of classes - optional -->
    </xsd:sequence>
</xsd:complexType>
```

```
<!-- definition of Inheritance type -->
<xsd:complexType name="pairType">
     <xsd:sequence>
          <xsd:sequence>
               <xsd:group ref="extraData" minOccurs="0" maxOccurs="unbounded"/>
               <!-- comments, descriptions, examples, userLabels relating to the inheritance go here -->
          </xsd:sequence>
          <xsd:element name="subclass" type="classRef"/>
          <xsd:element name="superclass" type="classRef"/>
     </xsd:sequence>
     <!-- inheritance is a pairwise relation, so inheritance declarations must be done in pairs -->
</xsd:complexType>
<xsd:complexType name="inheritance">
     <xsd:sequence>
          <xsd:element name="comment" type="comment" minOccurs="0" maxOccurs="unbounded"/>
          <xsd:element name="inheritancePair" type="pairType" minOccurs="0" maxOccurs="unbounded"/>
          <!-- comment: over all comments on the list of inheritances - optional -->
          <!-- inheritancePair: a tag followed by the subclass, superclass pair -->
     </xsd:sequence>
</xsd:complexType>
<!-- definition of comment type -->
<xsd:complexType name="comment">
     <xsd:simpleContent>
          <xsd:extension base="xsd:string">
               <xsd:attribute name="lang" type="xsd:language" use="optional"/>
          </xsd:extension>
     </xsd:simpleContent>
</xsd:complexType>
<!-- definition of packages  -->
<xsd:complexType name="packages">
     <xsd:sequence>
          <xsd:element name="package" minOccurs="0" maxOccurs="unbounded">
               <xsd:complexType>
```

```
                <xsd:attribute name="label" type="xsd:ID" use="required"/>
            </xsd:complexType>
        </xsd:element>
    </xsd:sequence>
</xsd:complexType>
<!-- the Ontology element defines the over-all structure of an ontology definition document -->
<xsd:element name="Ontology">
    <xsd:complexType>
        <xsd:all>
            <xsd:element name="ontologyOf" type="xsd:string" minOccurs="0"/>
            <xsd:element name="packages" type="packages" minOccurs="0"/>
            <xsd:element name="classes" type="classes" minOccurs="0"/>
            <xsd:element name="complexClasses" type="complexClasses" minOccurs="0"/>
            <xsd:element name="relations" type="relations" minOccurs="0"/>
            <xsd:element name="functions" type="functions" minOccurs="0"/>
            <xsd:element name="inheritance" type="inheritance" minOccurs="0"/>
            <!-- ontologyOf: a field for describing the general area being modelled by the ontology -->
            <!-- comment: free text comment on the Ontology -->
            <!-- classes:  list of declared classes -->
            <!-- complexClasses: list of defined complex classes -->
            <!-- relations: list of relations -->
            <!-- functions: list of functions -->
            <!-- inheritance: list of pairwise inheritance relations between classes -->
        </xsd:all>
        <xsd:attribute name="version" type="xsd:string" use="required"/>
        <xsd:attribute name="author" type="xsd:string" use="optional"/>
        <xsd:attribute name="lang" type="xsd:language" use="optional"/>
        <!-- version: version number of the definition document being composed-->
        <!-- author: optional name of creator(s) of the ontology document -->
        <!-- lang: optional declaration of main human language used in document -->
    </xsd:complexType>
</xsd:element>
</xsd:schema>
```

APPENDIX C

Example C1 – C6 includes the following classes:
Books, Persons, Strings, Countries, Languages, Corporations, Addresses, Dates, Integers;

and the following attributes:

Table II: Attributes for Examples C1 – C6

Domain	Attribute	Co-domain
Books	Author	Persons
	Title	Strings
	ISBN	Strings
	Library Catalogue Number	Strings
	Publisher	Corporations
	Date of Publication	Dates
Persons	Given Name	Strings
	Surname	Strings
	Country of Birth	Countries
	Countries of Citizenship	Countries
	Passport Number	Strings
	Address	Addresses
	Spouse	Persons
Countries	Name	Strings
	Bordering Countries	Set[Countries]
	Population	Integers
Corporations	Name	Strings
	Address	Addresses
Addresses	Building Number	Integers
	Street Name	Strings
	TownOrCity Name	Strings
	StateOrProvince Name	Strings
	Country	Countries
	Postal Code	Strings

The classes String and Integers are assumed to be classes equipped with fundamental views of type xsd:string and xsd:integer, respectively.

APPENDIX C1

I. View C1

Domain: Books

View: v = { (Author, v1), (Title, string_view), (ISBN, string_view), (Publisher, v4) ; Strict }

Domain: Persons

View: v1 = { (Given Name, string_view), (Surname, string_view), (Country of Citizenship, v1.3) ; Strict }

Domain: Countries

View: v1.3 = { (Name, string_view) ; Strict }

Domain: Corporations

View: v4 = { (Name, string_view), (Address, v4.2) ; Strict }

Domain: Addresses

View: v4.2 = { (TownOrCity Name, string_view), (Postal Code, string_view), (Country, v1.3) ; Strict }

II. XML Schema for View C1

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2000/10/XMLSchema"
elementFormDefault="qualified">

    <xsd:element name="Book" type="Book_View"/>

    <xsd:complexType name="Book_View">
        <xsd:all>
            <xsd:element name="author" type="Person_View"/>
            <xsd:element name="title" type="xsd:string"/>
            <xsd:element name="ISBN" type="xsd:string"/>
            <xsd:element name="publisher" type="Publisher"/>
        </xsd:all>
    </xsd:complexType>

    <xsd:complexType name="Person_View">
        <xsd:all>
            <xsd:element name="Given_Name" type="xsd:string"/>
            <xsd:element name="Surname" type="xsd:string"/>
            <xsd:element name="Country_of_Citizenship" type="Country"/>
        </xsd:all>
    </xsd:complexType>

    <xsd:complexType name="Country">
        <xsd:sequence>
            <xsd:element name="name" type="xsd:string"/>
        </xsd:sequence>
    </xsd:complexType>

    <xsd:complexType name="Publisher">
        <xsd:sequence>
            <xsd:element name="address" type="Address"/>
        </xsd:sequence>
        <xsd:attribute name="name" type="xsd:string" use="required"/>
    </xsd:complexType>

    <xsd:complexType name="Address">
        <xsd:all>
            <xsd:element name="TownOrCity_Name" type="xsd:string"/>
            <xsd:element name="Postal_Code" type="xsd:string"/>
            <xsd:element name="Country" type="Country"/>
        </xsd:all>
    </xsd:complexType>

</xsd:schema>
```

III. XML Document for Description of View C1

```
<?xml version="1.0" encoding="UTF-8"?>
<Book xmlns:xsi="http://www.w3.org/2000/10/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="C:\Unicorn\XML\booksViewStrict.xsd">

        <author>
            <Given_Name> Jonathan </Given_Name>
            <Surname> Swift </Surname>
            <Country_of_Citizenship>
                <name> Great Britain </name>
            </Country_of_Citizenship>
        </author>
        <title> Gulliver's Travels </title>
        <ISBN> 0-7932-95352 </ISBN>
        <publisher name="Kluwer">
            <address>
                <TownOrCity_Name> Boston </TownOrCity_Name>
                <Postal_Code> 02134 </Postal_Code>
                <Country>
                    <name> USA</name>
                </Country>
            </address>
        </publisher>

</Book>
```

APPENDIX C2

I. View C2

Domain: Books

v = { (author, v1), (Title, string_view), (ISBN, string_view), (Publisher, v4) ; Liberal }

Domain: Persons

v1 = { (Given Name, string_view), (Surname, string_view), (Country of Citizenship, v1.3) ; Strict }

Domain: Countries

v1.3 = { (Name, string_view) ; Strict }

Domain: Corporations

v4 = { (Name, string_view), (Address, v4.2) ; Liberal }

Domain: Addresses

v4.2 = { (TownOrCity Name, string_view), (Postal Code, string_view), (Country, v1.3) ; Strict }

II. XML Schema for View C2

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2000/10/XMLSchema"
elementFormDefault="qualified">

    <xsd:element name="Book" type="Book_View"/>

    <xsd:complexType name="Book_View">
        <xsd:all>
            <xsd:element name="author" type="Person_View" minOccurs="0" />
            <xsd:element name="title" type="xsd:string" minOccurs="0" />
            <xsd:element name="ISBN" type="xsd:string" minOccurs="0" />
            <xsd:element name="publisher" type="Publisher" minOccurs="0" />
        </xsd:all>
    </xsd:complexType>

    <xsd:complexType name="Person_View">
        <xsd:all>
            <xsd:element name="Given_Name" type="xsd:string"/>
            <xsd:element name="Surname" type="xsd:string"/>
            <xsd:element name="Country_of_Citizenship" type="Country"/>
        </xsd:all>
    </xsd:complexType>

    <xsd:complexType name="Country">
        <xsd:sequence>
            <xsd:element name="name" type="xsd:string"/>
        </xsd:sequence>
    </xsd:complexType>

    <xsd:complexType name="Publisher">
        <xsd:sequence>
            <xsd:element name="address" type="Address" minOccurs="0"/>
        </xsd:sequence>
        <xsd:attribute name="name" type="xsd:string" use="optional"/>
    </xsd:complexType>

    <xsd:complexType name="Address">
        <xsd:all>
            <xsd:element name="TownOrCity_Name" type="xsd:string"/>
            <xsd:element name="Postal_Code" type="xsd:string"/>
            <xsd:element name="Country" type="Country"/>
        </xsd:all>
    </xsd:complexType>

</xsd:schema>
```

III. XML Document for Description of View C2

```
<?xml version="1.0" encoding="UTF-8"?>
<Book xmlns:xsi="http://www.w3.org/2000/10/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="C:\Unicorn\XML\booksViewLiberal.xsd">

        <ISBN> 0-7932-95352 </ISBN>
        <title> Gulliver's Travels </title>

</Book>
```

APPENDIX C3

I. View C3

Domain: Books

v = { (author, v1), optional ; (Title, string_view), required ; (ISBN, string_view), required ; (Publisher, v4), optional }

Domain: Persons

v1 = { (Given Name, string_view), required ; (Surname, string_view), required ; (Country of Citizenship, v1.3), optional}

Domain: Countries

v1.3 = { (Name, string_view), required }

Domain: Corporations

v4 = { (Name, string_view), required ; (Address, v4.2), optional}

Domain: Addresses

v4.2 = { (TownOrCity Name, string_view), (Postal Code, string_view), (Country, v1.3) ; Strict }

II. XML Schema for View C3

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2000/10/XMLSchema"
elementFormDefault="qualified">

    <xsd:element name="Book" type="Book_View" />

    <xsd:complexType name="Book_View">
        <xsd:all>
            <xsd:element name="author" type="Person_View" minOccurs="0" />
            <xsd:element name="title" type="xsd:string" minOccurs="1" />
            <xsd:element name="ISBN" type="xsd:string" minOccurs="1" />
            <xsd:element name="publisher" type="Publisher" minOccurs="0" />
        </xsd:all>
    </xsd:complexType>

    <xsd:complexType name="Person_View">
        <xsd:all>
            <xsd:element name="Given_Name" type="xsd:string" minOccurs="1" />
            <xsd:element name="Surname" type="xsd:string" minOccurs="1" />
            <xsd:element        name="Country_of_Citizenship"        type="Country"
minOccurs="0" />
        </xsd:sll>
    </xsd:complexType>

    <xsd:complexType name="Country">
        <xsd:seqeunce>
            <xsd:element name="name" type="xsd:string"/>
            </xsd:sequence>
        </xsd:complexType>

        <xsd:complexType name="Publisher">
            <xsd:sequence>
                <xsd:element name="address" type="Address" minOccurs="0"/>
            </xsd:sequence>
            <xsd:attribute name="name" type="xsd:string" />
        </xsd:complexType>

        <xsd:complexType name="Address">
            <xsd:all>
                <xsd:element name="TownOrCity_Name" type="xsd:string"/>
                <xsd:element name="Postal_Code" type="xsd:string"/>
                <xsd:element name="Country" type="Country"/>
            </xsd:all>
        </xsd:complexType>

    </xsd:schema>
```

III. XML Document for Description of View C3

```
<?xml version="1.0" encoding="UTF-8"?>
<Book xmlns:xsi="http://www.w3.org/2000/10/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="C:\Unicorn\XML\booksViewMixed.xsd">

        <ISBN> 0-521-410304 </ISBN>
        <publisher name="Cambridge University Press"> </publisher>
        <title> Logic and Information </title>
        <author>
            <Given_Name> Keith </Given_Name>
            <Surname> Devlin </Surname>
        </author>

</Book>
```

APPENDIX C4

I. View C4

Domain: Books

v = { (author, v1), optional ; (Title, string_view), required ; (ISBN, string_view), required ; (Publisher, v4), optional ; Ordered }

Domain: Persons

v1 = { (Given Name, string_view), required ; (Surname, string_view), required ; (Country of Citizenship, v1.3) ; optional}

Domain: Countries

v1.3 = { (Name, string_view) ; required }

Domain: Corporations

v4 = { (Name, string_view), required ; (Address, v4.2), optional}

Domain: Addresses

v4.2 = { (TownOrCity Name, string_view), (Postal Code, string_view), (Country, v1.3) ; Strict ; Ordered}

II. XML Schema for View C4

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2000/10/XMLSchema"
elementFormDefault="qualified">

    <xsd:element name="Book" type="Book_View" />

    <xsd:complexType name="Book_View">
        <xsd:sequence>
            <xsd:element name="author" type="Person_View" minOccurs="0"/>
            <xsd:element name="title" type="xsd:string"/>
            <xsd:element name="ISBN" type="xsd:string"/>
            <xsd:element name="publisher" type="Publisher" minOccurs="0"/>
        </xsd:sequence>
    </xsd:complexType>

    <xsd:complexType name="Person_View">
        <xsd:all>
            <xsd:element name="Given_Name" type="xsd:string"/>
            <xsd:element name="Surname" type="xsd:string"/>
            <xsd:element name="Country_of_Citizenship" type="Country" minOccurs="0"/>
        </xsd:all>
    </xsd:complexType>

    <xsd:complexType name="Country">
        <xsd:sequence>
            <xsd:element name="name" type="xsd:string"/>
        </xsd:sequence>
    </xsd:complexType>

    <xsd:complexType name="Publisher">
        <xsd:sequence>
            <xsd:element name="address" type="Address" minOccurs="0"/>
        </xsd:sequence>
        <xsd:attribute name="name" type="xsd:string"/>
    </xsd:complexType>

    <xsd:complexType name="Address">
        <xsd:sequence>
            <xsd:element name="TownOrCity_Name" type="xsd:string"/>
            <xsd:element name="Postal_Code" type="xsd:string"/>
            <xsd:element name="Country" type="Country"/>
        </xsd:sequence>
    </xsd:complexType>

</xsd:schema>
```

III. XML Document for Description of View C4

```
<?xml version="1.0" encoding="UTF-8"?>
<Book xmlns:xsi="http://www.w3.org/2000/10/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="C:\Unicorn\XML\booksViewStrict.xsd">

        <author>
            <Country_of_Citizenship>
                <name> Great Britain </name>
            </Country_of_Citizenship>
            <Given_Name> Jonathan </Given_Name>
            <Surname> Swift </Surname>
        </author>
        <title> Gulliver's Travels </title>
        <ISBN> 0-7932-95352 </ISBN>
        <publisher name="Kluwer">
            <address>
                <TownOrCity_Name> Boston </TownOrCity_Name>
                <Postal_Code> 02134 </Postal_Code>
                <Country>
                    <name> USA</name>
                </Country>
            </address>
        </publisher>

</Book>
```

APPENDIX C5

I. View C5

Source: Books

v = { Choices: { (author, v1), (Title, string_view) }; {(ISBN, string_view) , (Library_Cat_Num, string_view) } Optional: {(Publisher, v4)} }

Source: Persons

v1 = { (Given Name, string_view), required ; (Surname, string_view), required ; (Country of Citizenship, v1.3), optional; Ordered}

Source: Countries

v1.3 = { (Name, string_view); required }

Source: Corporations

v4 = { (Name, string_view), required ; (Address, v4.2), optional}

Source: Addresses

v4.2 = { Choices: { (TownOrCity Name, string_view), (StateOrProvince Name, string_view) }; { (Country, v1.3) }; Optional: {(Postal Code, string_view)} }

II. XML Schema for View C5

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2000/10/XMLSchema"
elementFormDefault="qualified">

    <xsd:element name="Book" type="Book_View" />

    <xsd:complexType name="Book_View">

        <xsd:sequence>
            <xsd:choice>
                <xsd:element name="author" type="Person_View"  />
                <xsd:element name="title" type="xsd:string"  />
            </xsd:choice>
            <xsd:choice>
                <xsd:element name="ISBN" type="xsd:string"  />
                <xsd:element name="Lib_Cat" type="xsd:string" />
            </xsd:choice>

            <xsd:element name="publisher" type="Publisher"  minOccurs="0"/>

        </xsd:sequence>
    </xsd:complexType>

    <xsd:complexType name="Person_View">
        <xsd:sequence>
            <xsd:element name="Given_Name" type="xsd:string" minOccurs="1" />
            <xsd:element name="Surname" type="xsd:string" minOccurs="1" />
```

```
                <xsd:element name="Country_of_Citizenship" type="Country" minOccurs="0" />
            </xsd:sequence>
        </xsd:complexType>

        <xsd:complexType name="Country">
            <xsd:sequence>
                <xsd:element name="name" type="xsd:string"/>
            </xsd:sequence>
        </xsd:complexType>

        <xsd:complexType name="Publisher">
            <xsd:sequence>
                <xsd:element name="address" type="Address" minOccurs="0"/>
            </xsd:sequence>
            <xsd:attribute name="name" type="xsd:string" />
        </xsd:complexType>

        <xsd:complexType name="Address">

            <xsd:sequence>

                <xsd:choice>
                    <xsd:element name="TownOrCity_Name" type="xsd:string"/>
                    <xsd:element name="StateOrProvince_Name" type="xsd:string"  />
                </xsd:choice>
                <xsd:choice>
                    <xsd:element name="Country" type="Country"/>
                </xsd:choice>

                <xsd:element name="Postal_Code" type="xsd:string" minOccurs="0"/>

            </xsd:sequence>
        </xsd:complexType>

    </xsd:schema>
```

III. <u>XML Document for Description of View C5</u>

```
    <?xml version="1.0" encoding="UTF-8"?>
    <Book xmlns:xsi="http://www.w3.org/2000/10/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="C:\Unicorn\XML\booksViewRadioButton.xsd">

            <title> Gulliver's Travels </title>
            <Lib_Cat> QR342.63 </Lib_Cat>
            <publisher name="Kluwer">
                <address>
                    <TownOrCity_Name> Boston </TownOrCity_Name>
                    <Country>
                        <name> USA</name>
                    </Country>
                    <Postal_Code> 02134 </Postal_Code>
                </address>
            </publisher>

    </Book>
```

APPENDIX C6

I. View C6

Domain: Books

View: v = { (Authors, **set**(Person, v1)), (Title, string_view), (ISBN, string_view), (Publisher, v4) ; Strict }

Domain: Persons

View: v1 = { (Given Name, string_view), (Surname, string_view), (Country of Citizenship, v1.3) ; Strict }

Domain: Countries

View: v1.3 = { (Name, string_view) ; Strict }

Domain: Corporations

View: v4 = { (Name, string_view), (Address, v4.2) ; Strict }

Domain: Addresses

View: v4.2 = { (TownOrCity Name, string_view), (Postal Code, string_view), (Country, v1.3) ; Strict }

II. XML Schema for View C6

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2000/10/XMLSchema"
elementFormDefault="qualified">

    <xsd:element name="Books" type="List_Book_View" />

    <xsd:complexType name="List_Book_View">
        <xsd:sequence>
            <xsd:element name="book" type="Book_View" maxOccurs="unbounded"/>
        </xsd:sequence>
    </xsd:complexType>

<xsd:complexType name="Book_View">
        <xsd:sequence>
            <xsd:element name="authors" type="Set_Person_View"/>
            <xsd:element name="title" type="xsd:string"/>
            <xsd:element name="ISBN" type="xsd:string"/>
            <xsd:element name="publisher" type="Publisher"/>
        </xsd:sequence>
    </xsd:complexType>


    <xsd:complexType name="Set_Person_View">
        <xsd:sequence>
            <xsd:element name="author" type="Person_View" maxOccurs="unbounded"/>
        </xsd:sequence>
    </xsd:complexType>
    <xsd:complexType name="Person_View">
        <xsd:sequence>
            <xsd:element name="Given_Name" type="xsd:string"/>
            <xsd:element name="Surname" type="xsd:string"/>
            <xsd:element name="Country_of_Citizenship" type="Country"/>
        </xsd:sequence>
    </xsd:complexType>

    <xsd:complexType name="Country">
        <xsd:sequence>
            <xsd:element name="name" type="xsd:string"/>
        </xsd:sequence>
    </xsd:complexType>

    <xsd:complexType name="Publisher">
        <xsd:sequence>
            <xsd:element name="address" type="Address"/>
        </xsd:sequence>
        <xsd:attribute name="name" type="xsd:string" use="required"/>
    </xsd:complexType>

    <xsd:complexType name="Address">
        <xsd:sequence>
            <xsd:element name="TownOrCity_Name" type="xsd:string"/>
            <xsd:element name="Postal_Code" type="xsd:string"/>
            <xsd:element name="Country" type="Country"/>
        </xsd:sequence>
    </xsd:complexType>

</xsd:schema>
```

III. XML Document for Description of View C6

```
<?xml version="1.0" encoding="UTF-8"?>
<Books xmlns:xsi="http://www.w3.org/2000/10/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="C:\Unicorn\XML\booksViewContainer.xsd">

    <book>

        <authors>

            <author>
                <Given_Name> Chin </Given_Name>
                <Surname> Chang </Surname>
            </author>

            <author>
                <Given_Name>Jerome</Given_Name>
                <Surname> Keisler </Surname>
            </author>
        </authors>

        <title> Model Theory </title>
        <ISBN> 0-444-88054-2</ISBN>
        <publisher name="Elsevier">
            <address>
                <TownOrCity_Name> Amsterdam </TownOrCity_Name>
                <Postal_Code> 1000 </Postal_Code>
                <Country>
                    <name> Netherlands </name>
                </Country>
            </address>
        </publisher>
    </book>

    <book>
        <authors>

            <author>
                <Given_Name> Keith </Given_Name>
                <Surname> Devlin </Surname>
                <Country_of_Citizenship>
                    <name> USA </name>
                </Country_of_Citizenship>
            </author>

        </authors>
        <title> Logic and Information </title>
        <ISBN> 0-521-410304 </ISBN>
        <publisher name="Cambrigde University Press">
            <address>
                <TownOrCity_Name> Cambridge </TownOrCity_Name>
                <Postal_Code> CB2 1RP </Postal_Code>
                <Country>
                    <name> United Kingdom </name>
                </Country>
            </address>
        </publisher>
    </book>

</Books>
```

United States Patent & Trademark Office

Office of Initial Patent Examination -- Scanning Division

Application deficiencies found during scanning:

□ Page(s)__________ of ______________________________ were not present for scanning.
(Document title)

□ Page(s)__________ of ______________________________ were not present for scanning.
(Document title)

☑ ***Scanned copy is best available.*** Drawings.